基于翻译的信息隐藏分析检测

孟 朋[1],黄刘生[1,2],陈志立[1,2],杨 威[1,2],杨 明[1]
(1.中国科学技术大学计算机科学与技术系国家高性能计算中心,安徽合肥 230026;
2.中国科学技术大学苏州研究院,江苏苏州 215123)

摘 要: 基于翻译的信息隐藏(Translation-Based Steganography,TBS)是一类文本信息隐藏算法,它利用不同的翻译机对同一个句子翻译产生的结果一般不同这一特性,使用多台翻译机翻译同一段文本,最终译文的每个句子根据隐藏信息来选择不同翻译机的翻译结果以形成隐藏文本.这种方法基本保证了隐藏文本语法的正确性和语义的连贯性,传统的检测算法很难发现隐藏文本.本文研究发现,在知道 TBS 算法所使用的翻译机集合的条件下存在一种 TBS 检测算法,因此对 TBS 算法所使用的翻译机集合保密是 TBS 算法安全的关键.文章从理论上分析了检测算法的有效性,并给出了算法的过程和实验结果.另外,我们还给出了增强 TBS 安全性的方法.

关键词: 信息隐藏; 信息隐藏检测; 机器翻译; 机器可逆度; 机器倾向度; 机器生成度
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2010) 08-1748-05

Analysis and Detection of Translation-Based Steganography

MENG Peng[1], HUANG Liu-sheng[1,2], CHEN Zhi-li[1,2], YANG Wei[1,2], YANG Ming[1]
(1. *NHPCC, Department of Computer Science and Technology, USTC, Hefei, Anhui* 230027, *China*;
2. *Suzhou Institute for Advanced Study, USTC, Suzhou, Jiangsu* 215123, *China*)

Abstract: Translation-Based Steganography (TBS) is a class of text steganography. When translating the same sentence from a natural language to another by different translators, there are normally different translation results. Making use of this property, TBS chooses from different translation results of a sentence according to secrete messages to generate stego-text. Because TBS basically preserves syntactic correctness and semantic coherence, it is comparatively difficulty for traditional detections to detect this method. Our research shows that there exists a detection method for TBS when we know the Machine Translation (MT) set that being used by TBS. So it is crucial important to keep a secret of the MT set. This paper theoretically analyzes the effective of our detection method and gives the detection process and experiment results. Additionally, we propose a method to improve the security of TBS.

Key words: steganography; steganalysis; machine translation; degree of machine reversibility; degree of machine preference; degree of machine generated

1 引言

信息隐藏[1,2]是信息安全的主要研究方向之一,它研究如何将秘密信息嵌入已知载体,主要用于保密通信以及版权保护等.当前网络通信中保证信息安全的手段仍以传统的加密为主,但加密通信在被监视的情况下有被怀疑并破坏的弱点.信息隐藏技术掩盖了隐秘信息的存在,大大地增强了信息传输和存储的安全性.

由于文本媒体在互联网上的广泛使用,以文本为载体的信息隐藏技术越来越受到研究人员的关注.以文本为载体的信息隐藏从嵌入隐蔽信息的方式来分类,可以分为两大类:一是通过修改已有文本隐藏信息,二是通过生成近似自然语言的文本隐藏信息.对于第一类,可以通过改变文字间距[3],改变行间距[4],标点符号替换[5],以及同义词替换[6,7]等等.对于第二类,主要有采用概率上下文无关语法(PCFG)[8]、句子模版[9]或者采用机器翻译[10,11]等方式来生成含有隐藏信息的文本.

与图像和视频不同,自然文本的冗余空间很小,隐蔽信息的嵌入率较低,因此很难对其检测.已知检测算法主要针对 TLEX[12]、NICETEXT[9]、TEXTO[13]等嵌入率比较高的隐藏算法.文献[14]提出了一种基于词间相互关系的检测算法,对基于语法的文本信息隐藏算法进行

收稿日期:2009-07-04;修回日期:2009-08-28
基金项目:国家自然科学基金重大研究计划(No.90818005);国家自然科学基金(No.60773032、No.60703071);教育部博士点基金(No.2006CB303006);江苏省自然科学基金(No.BK2007060)

检测.文献[15]基于统计词的频度来检测 NICETEXT、TEXTO 等生成文本的隐藏算法.

现有检测算法所针对的隐藏算法嵌入率都比较高,它们完全不适合检测 TBS 算法.TBS 以句子为单位来隐藏信息,嵌入率很低,基本实现了隐藏文本语法的正确性和语义的连贯性,因此传统检测算法很难检测 TBS 算法.TBS 目前还没有有效的检测算法.本文研究显示,如果知道 TBS 算法所使用的翻译机集合,那么我们就可以实现对 TBS 算法的有效检测,并且通过理论分析验证了算法的有效性.因此对 TBS 算法所使用的翻译机集合进行保密将对 TBS 算法的安全性具有重要意义.

2　基于翻译的信息隐藏算法介绍

信息隐藏顶级会议 IH2005 首次提出 TBS 算法 Lost in Translation (LiT)[10], 文献[11]中提出的 Lost in Just the Translation (LiJtT)算法是对文献[10]中提出的 LiT 算法的改进.

2.1　LiT 算法

LiT 算法[10]原理如下:由于机器翻译"噪声"的存在,不同的翻译机对同一个句子翻译的结果一般不同.可以根据翻译机翻译结果质量的不同,将翻译机组织成一棵 Huffman 树.例如,有 3 台翻译机 Google[16]、Systran[17]和 Linguatec[18],可以假设 Google 代表 1,Systran 代表 01,Linguatec 代表 00,对于一篇待翻译文本,使用 3 台翻译机对其翻译,每个句子就有 3 个结果可以选择,根据需要隐藏的信息和翻译机对应的 Huffman 树来选择相应的句子形成隐藏文本.

发送端 Alice 和接收端 Bob 共享翻译机信息以及翻译机的组织形式等信息,当 Bob 收到隐藏文本和原文本,将原文本采用和发送端同样的方式进行翻译,得到多篇翻译文本,然后和接收到的隐藏文本对比,判断接收到的隐藏文本的每个句子来自哪一个翻译机,从而根据翻译机的 Huffman 树提取出对应的隐藏信息.

2.2　LiJtT 算法

LiJtT 算法[11]根据当前需要隐藏的信息和多个翻译结果的 Hash 值来选择相应的句子.LiJtT 算法首先对不同的翻译结果进行 Hash,选择 Hash 值的一段作为隐藏信息段,然后选择 Hash 值的隐藏信息段和当前需要隐藏的信息对应的句子形成隐藏文本.当接收端收到隐藏文本,对每一个句子使用和发送端相同的密钥进行 Hash,取得 Hash 值的隐藏信息段,即获得隐藏信息. LiJtT 省略了原文本的传递,降低了传输带宽的需求,提高了系统的安全性,但算法实现起来比较困难.

3　概念定义

为了方便后面的分析介绍,定义以下概念:

定义 1　翻译文本:普通文本经过某一台翻译机翻译生成的文本.

定义 2　隐藏文本:TBS 算法生成的含有隐藏信息的文本.

定义 3　自然文本:除了翻译文本和隐藏文本以外,任何一篇没有经过特意处理的自然语言文本都称为自然文本.

翻译文本和自然文本合称为正常文本.

定义 4　相似度:句子 S_1 与句子 S_2 的相似度定义为:S_1 中所有 2 元组(相邻单词)在 S_2 中出现的次数除 S_1 以中所有的 2 元组数.如果 S_1 中一个 2 元组在 S_2 中出现多次,则只计算一次.

例如:

S_1:I think our human brains are an evolutionary accident.

S_2:I believe that our human brain is an evolutionary accident.

S_2 与 S_1 的相似度为:3/9;S_1 与 S_2 的相似度为:3/8.

定义 5　机器可逆度(Degree of Machine Reversibility,DMR):使用翻译机 MT 将句子 S 翻译为另外一种语言(如英语 > 德语)的句子 S',然后使用同一翻译机将句子 S' 翻译回源语言句子(如德语 > 英语)S'',那么句子在翻译机 MT 的机器可逆度定义为:S'' 与 S' 的相似度.

定义 6　机器倾向度(Degree of Machine Preference,DMP):设有文本段 T 和一个翻译机集合 MTS,T 含有 N 个句子,如果其中 n_1 个句子在翻译机 M_1 的机器可逆度大于或等于在 MTS 中所有其他翻译机的机器可逆度,那么在 MTS 中,T 对 M_1 的机器倾向度为 n_1/N.

定义 7　机器生成度(Degree of Machine Generated,DMG):对于某类文本和一个翻译机集合 MTS,此类文本相对于 M_1 的机器生成度定义为:在 MTS 中,文本对翻译机 M_1 的机器倾向度大于或等于 0.5 的概率.

4　算法思想

检测算法的目的,是要区分出正常文本和隐藏文本.我们首先研究正常文本和隐藏文本的特性.正常文本分别研究自然文本和经过 Google、Systran 和 Linguatec 等翻译机生成的翻译文本;隐藏文本研究使用上述 3 台翻译机的 TBS 算法生成的文本.本文所研究的文本皆为英文文本,从以下 4 个方面来分析.

4.1　正常文本的机器可逆度分布规律

从图 1 可以看出,自然文本的句子在 Google 翻译机的机器可逆度,除某些特殊值(如 0 和 1)之外,近似满足正态分布.实际上,自然文本以及翻译文本皆具有类

似的性质,我们假设:正常文本的同一类句子(自然语言句子或某一台翻译机生成的句子)在同一台翻译机上的机器可逆度为正态分布.

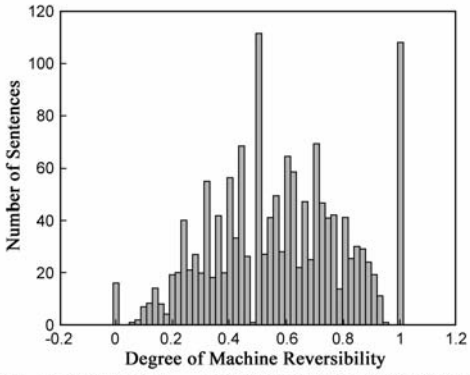

图1 自然文本在Google翻译机的机器可逆度分布图

我们统计自然语言句子和使用 Google、Systran 和 Linguatec 翻译机生成的句子在 Google、Systran 和 Linguatec 翻译机的机器可逆度的平均值和方差,数据如表 1 所示:(NEN、GEN、SEN、LEN 依次表示自然文本和分别经过 Google、Systran 和 Linguatec 翻译机生成的英文文本).

表 1 不同类型句子在不同翻译机的 DMR 的平均值和方差

DMR		Google	Systran	Linguatec
NEN	AVG	0.5781	0.3803	0.3524
	VAR	0.0537	0.0413	0.0447
GEN	AVG	0.7936	0.4622	0.3767
	VAR	0.0449	0.0471	0.0466
SEN	AVG	0.3910	0.5690	0.3269
	VAR	0.0424	0.0558	0.0391
LEN	AVG	0.4013	0.4030	0.6134
	VAR	0.0442	0.0454	0.0581

从表 1 可以看出,自然语言句子在 Google 翻译机的平均机器可逆度明显高于其他两台翻译机.这是因为 Google 翻译机是基于统计的翻译机,而且有庞大的数据库支持,所以翻译出来的文本更接近于自然语言.对于翻译生成的文本,3 台翻译机生成的文本都在生成自身的翻译机有最高的平均机器可逆度.这是因为机器翻译的文本会留下翻译机的"烙印",而翻译机会对自身留下的"烙印"有较高的识别作用.

4.2 正常文本的机器生成度计算

根据 4.1 的假设和数据,可以计算任一类文本在翻译机集合 Google、Systran 和 Linguatec 中相对每一台翻译机的机器生成度.例如计算自然文本相对 Google 翻译机的机器生成度,过程如下所示.

首先,分别计算自然文本句子在 Google 的机器可逆度大于 Systran 和大于 Linguatec 的概率.自然文本句子在 Google 的机器可逆度大于 Systran 的概率为:

$$P_{ngs}^{DMR}=\int_{-\infty}^{\infty}\frac{1}{\sigma_{ns}\sqrt{2\pi}}e^{-\frac{(t-\mu_{ns})^2}{2\sigma_{ns}^2}}\int_{t}^{\infty}\frac{1}{\sigma_{ng}\sqrt{2\pi}}e^{-\frac{(x-\mu_{ng})^2}{2\sigma_{ng}^2}}dx\,dt \tag{1}$$

P_{ngs}^{DMR}下标第一个字母代表文本类型,n、g、s、l 和 h 依次代表自然文本、经过 Google、Systran 和 Linguatec 翻译的文本和隐藏文本,第二和第三个字母代表翻译机,g 代表 Google 翻译机,s 代表 Systran 翻译机,l 代表 Linguatec 翻译机,σ_{ns}和μ_{ns}下标第一字母表示文本类型,第二字母表示翻译机,σ_{ns}和μ_{ns}分别表示自然文本在 Systran 翻译机的机器可逆度的标准差和平均值.

同理可以计算自然文本句子在 Google 翻译机的机器可逆度大于在 Linguatec 翻译机的概率 P_{ngl}^{DMR}.

对于含有 n 个句子的自然文本,相对 Google 翻译机的机器生成度为:

$$P_{ng}^{DMR}\sum_{i=\frac{n}{2}}^{n}\binom{n}{i}((P_{ngs}^{DMR}P_{ngl}^{DMR})^{i}(1-P_{ngs}^{DMR}P_{ngl}^{DMR})^{n-i}) \tag{2}$$

同理,可以计算文本相对 Systran 和 Linguatec 翻译机的机器生成度.当文本大小为 54 个句子时,根据表 1 数据,计算结果如下:

表 2 正常文本相对不同翻译机的机器生成度

DMG	Google	Systran	Liuguatec
NEN	0.8646	0	0
GEN	1	0	0
SEN	0	0.8495	0
LEN	0	0	0.8283

从表 2 可以看出,通过计算机器生成度可以明显的区分出不同翻译机翻译生成的文本.计算还显示,自然文本相对 Google 翻译机的机器生成度和翻译文本相对产生自身翻译机的机器生成度都会随着文本的增大而升高.

4.3 隐藏文本的机器生成度计算

因为隐藏文本的句子来自不同的翻译机,其机器生成度无法采用式(2)计算.在隐藏算法使用 3 台翻译机的条件下,我们假设隐藏文本中来自三台翻译机的句子各占 1/3,隐藏文本的机器生成度需对隐藏文本中来自不同翻译机的句子分别计算.例如计算含有 n 个句子的隐藏文本相对 Google 翻译机的机器生成度,计算公式如下:

$$\begin{aligned}P_{hg}^{DMG}=&\sum_{s=\frac{n}{2}}^{n}\sum_{i=\max(0,s-\frac{2n}{3})}^{\frac{n}{3}}\sum_{j=\max(0,s-i-\frac{n}{3})}^{\min(\frac{n}{3},s-i)}\\&\left(\binom{\frac{n}{3}}{i}(P_{ggs}^{DMR}P_{ggl}^{DMR})^{i}(1-P_{ggs}^{DMR}P_{ggl}^{DMR})^{\frac{n}{3}-i}\right.\\&\binom{\frac{n}{3}}{j}(P_{sgs}^{DMR}P_{sgl}^{DMR})^{j}(1-P_{sgs}^{DMR}P_{sgl}^{DMR})^{\frac{n}{3}-j}\\&\left.\binom{\frac{n}{3}}{k}(P_{lgs}^{DMR}P_{lgl}^{DMR})^{k}(1-P_{lgs}^{DMR}P_{lgl}^{DMR})^{\frac{n}{3}-k}\right)\end{aligned}$$

其中,$k=s-i-j$ (3)

同理可以计算隐藏文本相对其他两台翻译机的机器生成度.根据表 1 数据,计算结果显示:当文本大于 54 个句子时,隐藏文本相对于 3 台翻译机的机器生成度都小于 0.006,且相对于每一台翻译机的机器生成度都随着文本增大而减小.

4.4　用机器倾向度检测 TBS 的有效性分析

通过以上分析计算可知:在一个特定的翻译机集合中,正常文本都会相对于某一台翻译机有较高的机器生成度,而且文本越大,机器生成度也越大.而隐藏文本相对每一台翻译机的机器生成度都很小,同时文本越大,机器生成度越小.

如果以是否出现相对某一台翻译机的机器倾向度大于或等于 0.5 作为正常文本和隐藏文本的分类标准,对于一篇含有 54 个句子以上的文本,从上面的计算结果简单分析可知,隐藏文本的分辨准确率高达 98.2% 以上,而正常文本分辨准确率可以高达 82.8% 以上.因此,使用机器倾向度来分辨正常文本和隐藏文本,可以达到非常理想的效果.

实际检测过程,我们计算一篇文本在一个翻译机集合中对每个翻译机的机器倾向度,形成一个机器倾向度数组,通过 SVM 分类器[19]对文本进行分类,可以达到更加理想的效果.

5　实验和 TBS 安全性改进

实验需要训练文本和测试文本,我们对大小为 40、60 和 80 个句子的文本分别进行训练和测试.训练文本和测试文本皆包含自然文本,由德文分别经过 Google、Systran 和 Linguatec 翻译机翻译生成的英文文本和 TBS 算法生成的隐藏文本(TBS 算法使用三台翻译机:Google、Systran 和 Linguatec,隐藏过程为将德文文本翻译为英文文本).

5.1　检测过程和实验结果

假设隐藏算法所使用的翻译机集合为 MTS,其中含有 m 台翻译机.我们判断一段文本是否为 TBS 生成的隐藏文本,步骤如图 2 所示.

表 3 的训练文本列表示训练 SVM-lib 所使用的文本数.测试文本列表示实验测试的文本数,错误列表示算法错误分类的文本数.准确率等于正常文本和隐藏文本中所有错误分类的文本段数之和除以正常文本和隐藏文本的测试文本段数之和.

表 3　基于翻译的信息隐藏检测算法实验结果数据

大小	文本类型	训练文本	测试文本	错误	准确率(%)
40	正常	200	1000	76	89.36
	隐藏	50	250	57	
60	正常	200	1000	53	93.28
	隐藏	50	250	31	
80	正常	200	1000	17	96.64
	隐藏	50	250	25	

步骤 1:对给定的待测文本段 T 进行分句,得到一个句子序列;$(S_1, S_2, \cdots, S_n)$

步骤 2:将所有句子通过 MTS 中所有翻译机翻译为另外一种和源语言接近的语言的句子(如英语到德语),翻译机翻译生成的文本记为 $M_i(S_1', S_2', \cdots, S_n')$, $i = 1, \cdots, m$.

步骤 3:将 $M_i(S_1', S_2', \cdots, S_n')$, $i = 1, \cdots, m$ 再用翻译机翻译生成源语言的文本(如德语到英语)$M_i(S_1'', S_2'', \cdots, S_n'')$.

步骤 4:根据$(S_1, S_2, \cdots, S_n)$和 $M_i(S_1'', S_2'', \cdots, S_n'')$, $i = 1, \cdots, m$ 计算每个句子在 MTS 中所有翻译机上的机器可逆度.

步骤 5:计算文本对翻译机集合 MTS 中所有翻译机的机器倾向度,形成机器倾向度数组.

$DMP[DMP_1, DMP_2, \cdots, DMP_m]$

步骤 6:将机器倾向度数组调整为适合 SVM-lib[19]分类的数据格式.

步骤 7:根据分类器 SVM-lib 输出判断 T 为正常文本还是隐藏藏文本.

图 2　基于翻译的信息隐藏检测算法步骤

5.2　TBS 安全性分析

从表 3 可以看出,我们的检测算法可以有效的检测 TBS 算法,而且文本越大,检测结果越准确,这和理论分析相吻合.当前的算法假设知道隐藏算法所使用的翻译机的集合,如果不知道隐藏算法所使用的翻译机的集合,检测算法就要构造一个尽可能大的翻译机集合,使得检测算法所使用的翻译机集合尽量包含隐藏算法所使用的翻译机集合,这样会使得检测准确度大大降低.因此对 TBS 算法所使用的翻译机集合进行保密是 TBS 算法安全的关键.

另外,本文的检测算法是根据文本的机器倾向度来区分正常文本和隐藏文本.正常文本都会对某一台翻译机有较高的机器倾向度,隐藏文本的句子因为来自不同的翻译机,对所有翻译机(TBS 算法所使用的翻译机)的机器倾向度都较低.如果刻意增加隐藏文本中来自某一台翻译机 M 的句子的频率,那么隐藏文本对翻译机 M 的机器倾向度就会高于其他机器,形成的机器倾向度数组也就接近于正常文本.例如,控制隐藏文本中来自 Google 翻译机的句子所占比例为 50%,使用上述检测算法对大小为 40 个句子的隐藏文本的检测准确率为 61%;如果隐藏文本中来自 Google 翻译机的句子所占比例为 67%,那么对隐藏文本的检测准确率降为 6%.因此可以通过设置 TBS,使得隐藏文本中某一台翻译机的句子出现频率远高于其他翻译机来增加 TBS 的安全性.

6　总结

本文对 TBS 算法进行分析检测.首先假设我们知道 TBS 算法所使用的翻译机集合,然后在此基础上给

出一个有效的 TBS 检测算法,以此说明对 TBS 算法所使用的翻译机集合进行保密是 TBS 算法安全的关键.我们给出了机器可逆度、机器倾向度和机器生成度的概念,研究了自然文本以及机器翻译文本的特点,统计出了不同类型文本的机器可逆度分布规律.通过计算不同类型文本的机器生成度,从理论上说明了我们的检测算法的可靠性.另外,本文的研究思想和研究成果对机器翻译自动评测也有积极意义.

参考文献:

[1] K Bennett. Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text[R]. Purdue University, 2004 – 13.

[2] Petitcolas, FAP, Anderson, RJ, Kuhn, MG. Information hiding-A survey[J]. Proceedings of the IEEE, 1999, 87(7): 1062 – 1078.

[3] J T Brassil, S Low, N F Maxemchuk. Copyright protection for electronic distribution of text documents[J]. Proceedings of the IEEE, 1999, 87(7): 1181 – 1196.

[4] J Brassil, S Low, N Maxemchuk, L O'Garman. Electronic marking and identification techniques to discourage document copying[J]. IEEE Journal on Selected Areas in Communications, 1995, 13(8): 1495: 1504.

[5] C Weibing, D Guanzhong, X Yu, M Dejun. Tchnology of information hiding based on text document[J]. Application Research Of Computers, 2003, 20(10): 39 – 41.

[6] Bergmair R. Towards Linguistic Steganography: A Systematic Investiga-tion of Approaches Systems and Issues[R]. Vienna, Austria, University of Derby, 2004.

[7] Atallah M J, McDonough C J, Raskin V, et a1. Natural language processing for information assurance and security: an overview and implementations[A]. Proceedings New Security Paradigm Workshop[C]. New York: ACM, 2000. 51 – 65.

[8] Wayner, P. Mimic functions[J]. Cryptologia 1992, 16(3): 193 – 214.

[9] Chapman M, G Davida. Hiding the hidden: a software system for concealing ciphertext as innocuous text[A]. In Proceedings of the International Conference on Information and Communication Security. 1997, Lecture Notes in Computer Sciences 1334 [C]. Berlin: Springer, 1997. 333 – 345.

[10] Christian Grothoff, Krista Grothoff, Ludmila Alkhutova, Ryan Stutsman, Mikhail J Atallah. Translation-based steganography [A]. In Proceedings of Information Hiding Workshop (IH 2005)[C]. Berlin: Springer, 2005. 213 – 233.

[11] Ryan Stutsman, Mikhail Atallah, Christian Grothoff, Krista Grothoff. Lost in just the translation[A]. In Proceedings of the 2006 ACM Symposium on Applied Computing[C]. New York: ACM, 2006. 338 – 345.

[12] W instein K. Lexical Steganography Through Adaptive Modulation of the Word Choice Hash[EB/OL]. http://www.imsa.edu/ ~ keithw/tlex, 1999 – 03.

[13] K Maher. TEXTO[CP]. ftp://ftp.funet.fi/pub/crypt/steganography/texto.tar.gz, 2008-11.

[14] Zhili Chen, Liusheng Huang, et al. Linguistic steganography detection using statistical characteristics of correlations between words[A]. Information Hiding 2008, LNCS[C]. Berlin: Springer, 5284, 2008. 224 – 235.

[15] Zhili Chen, Liusheng Huang, et al. Effective Linguistic Steganography Detection[A]. In Proceedings of IEEE 8th International Conference on Computer and Information Technology Workshops[C]. Australia : CIT Workshops, 2008. 224 – 229.

[16] Google. Google translation[CP/OL]. http://www.google.com/language_tools, 2008 – 11.

[17] Systran. Systran translation[CP/OL]. http://www.systransoftware.cn/, 2008 – 11.

[18] Linguatec. Linguatec translation[CP/OL]. http://www.linguatec.de, 2008 – 11.

[19] Chih-Chung Chang, Chih-Jen Lin. LIBSVM: a library for support vector machines[CP]. http://www.csie.ntu.edu.tw/ ~ cjlin/libsvm, 2001.

作者简介:

孟 朋 男,1983 年生于安徽萧县,中国科学技术大学计算机系博士研究生,研究方向为信息安全.

E-mail: mengpeng@mail.ustc.edu.cn

黄刘生 男,1957 年出生,教授,博士生导师,主要研究领域为信息安全,高性能算法,分布式计算等.